On the Partitioning of Servers in Queueing Systems during Rush Hour

Bin Hu • Saif Benjaafar

Department of Operations and Management Science, Ross School of Business, University of Michigan at Ann Arbor, 701 Tappan St., Ann Arbor, Michigan 48109, USA
Division of Industrial and Systems Engineering, Department of Mechanical Engineering, University of Minnesota at Minneapolis, 111 Church Street S.E., Minneapolis, Minnesota 55455, USA

hub@umich.edu • saif@umn.edu

This paper is motivated by two phenomena observed in many queueing systems in practice. The first is the partitioning of server capacity among different customers based on their service time requirements. The second is rush hour demand where a large number of customers arrive over a short period of time followed by few or no arrivals for an extended period thereafter. We study a system with multiple parallel servers and multiple customer classes. The servers can be partitioned into server groups, each dedicated to a single customer class. The system experiences rush hour demand when most arrivals occur. We compare the performance of the system with and without server partitioning during rush hour and address three basic questions. (1) Is partitioning beneficial to the system? (2) Is it equally beneficial to all customer classes? (3) If it is implemented, what is an optimal partition? We provide remarkably simple answers to all three questions and we do so without making any distributional assumptions on service times. Our results are also independent of the number of customers that arrive during rush hour. We present numerical results to illustrate the benefit of partitioning.

Keywords: Server partitioning, multi-server queueing systems, multiple demand classes, rush hour demand

1. Introduction

This paper is motivated by two phenomena observed in many queueing systems in practice. The first is the partitioning of server capacity among different customers based on their service requirements. For example, in systems with parallel servers, it is not uncommon to dedicate a subset of the servers to customers with the shortest service times. The classic example is of course the express lane(s) found in grocery stores, but it is also observed
elsewhere including banks, airports, government offices, and call centers. The second phenomenon is the arrival of a large number of customers over a short period of time followed by few or no arrivals for an extended period thereafter. This situation is observed in systems subject to rush hour demand, such as toll booths on highways during peak hours, fast-food restaurants during lunch time, concession stands in stadiums during intermissions, customs and immigration controls at airports following an international flight, and many others. The common feature to these examples is the almost simultaneous arrivals of a large number of customers followed by few or no additional arrivals until the next rush hour. In such systems, the primary concern is the waiting time that customers experience during this period since there is typically ample capacity at other time.

Despite the prevalence of both phenomena in practice, they appear not to have been sufficiently studied in the literature. In particular, there are few results regarding how to best partition servers among customer classes and whether or not this partitioning is beneficial. Similarly, there are relatively few results for systems subject to rush hour demand, particularly with respect to how capacity should be managed during this period.

We provide remarkably simple answers to all three questions and we do so without making any distributional assumptions on service times. First, we show that partitioning can be indeed beneficial to the system as a whole. Second, we show that there does not exist a partitioning of servers to customers under which all customer classes are better off (although there exists a partitioning under which all customers are indifferent between the partitioned and unpartitioned system). Third, we show that there exists a unique partitioning that optimizes the performance of the system as a whole and we provide a closed form expression for the number of servers allocated to each class under this optimal partition. We present numerical results documenting the benefits of partitioning under different scenarios.

There is a rich literature in queueing theory that compares partitioned versus unpartitioned systems. The partitioned system is often in the form of multiple single server queues with each queue serving an independent stream of customers who arrive continuously over time with stochastic inter-arrival times. The unpartitioned system on the other hand is in the form of a single multi-server queue from which all customers are served on a first-come, first-served basis; see for example Kleinrock (1976), Rothkopf and Rech (1987), Smith and Whitt (1981), Whitt (1992, 1999), Benjaafar (1995), and Benjaafar et al. (2005). An important insight from this literature is that the unpartitioned system is superior as long as all customers have identical service time distributions, but not necessarily so when customer
streams have different service time requirements. The setting for this literature is different from ours in that it assumes steady state operation over an infinite horizon and continuous customer arrivals. Most of this literature also assumes that the set of servers associated with each demand stream in the partitioned system either consists of a single server or is exogenously determined. A notable exception is Whitt (1999) who presents a heuristic procedure for assigning servers to each customer stream.

One of the benefits of partitioning, both in the queueing literature and in our model, is protecting customers with short processing times from experiencing delays due to customers with long processing times. This of course can be achieved without resorting to partitioning but by assigning different priorities to different customer streams or classes. However, having customers wait in a single queue and be sequenced based on a priority scheme can be unpractical in some settings or not acceptable to certain customers in others; see Rothkopf and Rech (1987) for an illuminating discussion on this topic. Hence, partitioning can be viewed as an alternative to priority sequencing.

The rest of this paper is organized as follows. In Section 2, we describe our model for systems with and without partitioning. In Sections 3 and 4, we describe our main theoretical results. In Section 5, we provide numerical results illustrating the benefit of partitioning. In Section 6, we offer some concluding comments.

2. Problem Description and Preliminary Results

We consider a queueing system consisting of $n$ identical servers and $l$ customer classes. Service times within each class are assumed to be independent and identically distributed with mean $E(S_i)$ for customer class $i$, $i = 1, \cdots, l$ where $S_i$ is a random variable denoting service time for customer class $i$. Without loss of generality, we assume that $E(S_1) \leq E(S_2) \leq \cdots \leq E(S_l)$. We assume that the probability that a customer is of type $i$ is $p_i$, with $0 \leq p_i \leq 1$ and $p_1 + \cdots + p_l = 1$. We consider two scenarios, one in which the servers are partitioned among the customer classes with each class $i$, $i = 1, \cdots, l$, assigned $k_i$ servers where the integer $k_i > 0$ and $k_1 + \cdots + k_l = n$. We refer to this system as the partitioned system. The other scenario is one in which all servers are accessible to all customers. We refer to this system as the unpartitioned system. Upon arrival, a customer chooses a server among those assigned to his class and waits in the queue of that server until the server becomes available. We assume that no jockeying is allowed so that customers do not switch
queues once they have joined one. Within each queue, we assume that customers are served on a first-come, first-served basis. Once a customer receives service, the customer leaves the system. Of course in the unpartitioned system, a customer can choose any server. For both the partitioned and unpartitioned systems, we assume that customers choose to join the queue with the fewest customers. This means that in the absence of partitioning a customer does not know the type of other customers who are already in the system.

We are concerned with a rush hour regime of operation whose beginning is marked by the almost simultaneous arrival of \( m \) customers to the system, with \( m \) being much greater than the number of servers (\( m >> n \)). We assume that no further arrivals occur until these \( m \) customers have cleared the system. This is obviously an approximation of rush hour phenomena in practice since arrivals typically continue to occur beyond the initial arrival rush. However, it is arguably a reasonable approximation when \( m \) is much greater than the number of additional arrivals that occur after the initial rush. We let \( m_i = p_i m \) denote the number of customers of type \( i \) and also assume that \( m_i >> n \). Since customers choose to join the server with the shortest queue immediately upon arrival, the length of the queues dedicated to class \( i \) would each have \( m_i/k_i \) customers (including the one in service) once all \( m_i \) customers have arrived. We shall treat \( m_i/k_i \) as an integer since \( m_i >> k_i \). In a system without service partitioning, the length of all queues is equal to \( m/n \), which we also treat as an integer.

We use expected time customers spend in the system as our measure of performance. However, our results are applicable to related measures such as expected waiting time in the queue or expected number of customers in the queue or the system. For the unpartitioned system, the expected time a customer spends in the system (regardless of his class) is given by

\[
E(W_u) = \sum_{i=1}^{l} p_i E(S_i) \left( \frac{1 + 2 + \cdots + m/n}{m/n} \right) = \left( \frac{1}{2} + \frac{m}{2n} \right) \sum_{i=1}^{l} p_i E(S_i). \tag{1}
\]

Similarly, for the partitioned system, the expected time in system for customers of type \( i \) is given by

\[
E(W_{p_i}(k_i)) = \left( \frac{1}{2} + \frac{p_i m}{2k_i} \right) E(S_i), \tag{2}
\]

and the expected time in system for an arbitrary customer is

\[
E(W_p(k)) = \sum_{i=1}^{l} p_i E(W_{p_i}(k_i)), \tag{3}
\]
where \( k = (k_1, \ldots, k_l) \) is a vector specifying the number of servers dedicated to each class. Although the above performance measures depend on the parameter \( m \), we show in the next two sections that our main results in Theorems 1 and 3 are in fact independent of \( m \). We will also show that they are independent of distributional assumptions about service times.

3. On the Fairness of Partitioning

In this section, we address the question of fairness. Namely, is it possible to find a server partitioning scheme under which all customer classes are better off than in the unpartitioned system?

Theorem 1 Let

\[
\mathbf{k}^w = (k_1^w, \ldots, k_l^w) = \left( \frac{p_1 E(S_1)}{\sum_{i=1}^l p_i E(S_i)} n, \ldots, \frac{p_l E(S_l)}{\sum_{i=1}^l p_i E(S_i)} n \right).
\]

Then, the following equivalence holds for all \( i \):

\[
E(W^p_i(k_i)) \leq E(W^u) \text{ if and only if } k_i \geq k_i^w.
\]

Therefore the unique solution to the system of inequalities

\[
\begin{align*}
E(W^p_1(k_1)) &\leq E(W^u) \\
\vdots \\
E(W^p_l(k_l)) &\leq E(W^u)
\end{align*}
\]

is

\[
\begin{align*}
k_1 = k_1^w \\
\vdots \\
k_l = k_l^w
\end{align*}
\]

for which \( E(W^p_i(k_i^w)) = E(W^u) \) for all \( i \).

The proof for Theorem 1 is straightforward and we omit it. Theorem 1 shows that a partitioned system can never simultaneously improve the performance of all customer classes. Hence, an improvement achieved by any customer class comes only at the expense of other classes. It is possible for a partitioned system to provide the same level of performance for each class as an unpartitioned system. However, this is the case if and only if each class is allocated a number of servers proportional to its workload, that is \( k_i = k_i^w \). We refer to such
an allocation as the *workload-proportional* allocation. Note that a workload-proportional allocation is feasible only if the \( k^w_i \) are integer valued. If not, then no allocation exists under which the partitioned and unpartitioned systems are equivalent.

Although workload allocation (if feasible) is the only allocation that guarantees that each class is not worse off with partitioning, it is not the only allocation that could provide the same overall expected time in system as the unpartitioned system.

**Theorem 2** Let

\[ k^m = (k^m_1, \cdots, k^m_l) = (p_1 n, \cdots, p_l n). \]  

(5)

Then \( E(W^p(k^m)) = E(W^u) \). Furthermore, for any \( 1 \leq j \leq l \) such that \( E(S_j) \leq \sum_{i=1}^l p_i E(S_i) \), we have \( k^m_j \geq k^w_j \). Otherwise, if \( E(S_j) \geq \sum_{i=1}^l p_i E(S_i) \), we have \( k^w_j \geq k^m_j \).

The proof is also straightforward and we again omit it. Theorem 2 shows that an allocation proportional to the population of each class leads to the same overall system performance as the unpartitioned system. This allocation is of course feasible only if it yields integer-valued \( k^m_i \)'s. We refer to this allocation as the *mix-proportional* allocation. As we can see, a mix-proportional allocation favors customer classes with a relatively short services times (classes whose mean service times is smaller than the overall mean service time); these customers receive more servers than they would under the workload-proportional allocation. Consequently the performance of other customer classes suffers. We will use this property in the next section to further characterize the optimal allocation.

4. **Optimal Partitioning**

In this section, we address the question of whether or not partitioning can improve the performance of the system, even though it may not improve the performance of all customers, and if so what is the optimal way to partition servers among different classes.

**Theorem 3** Let

\[ k^* = (k^*_1, \cdots, k^*_l) = \left( \frac{p_1 \sqrt{E(S_1)}}{\sum_{i=1}^l p_i \sqrt{E(S_i)}} n, \cdots, \frac{p_l \sqrt{E(S_l)}}{\sum_{i=1}^l p_i \sqrt{E(S_i)}} n \right). \]  

(6)

1. The minimum

\[ \min_{\sum_{i=1}^l k_i = n} E(W^p(k)) = \frac{1}{2} \sum_{i=1}^l p_i E(S_i) + \frac{m}{2n} \left( \sum_{i=1}^l p_i \sqrt{E(S_i)} \right)^2 \]  

(7)

is attained with the unique vector \( k^* \).
2. For each $1 \leq j \leq l$, if $E(S_j) \leq \sum_{i=1}^{l} p_i E(S_i)$ then $k^*_j \geq k^w_j$; otherwise $k^*_j \geq k^m_j$. In general, $k^*_j \geq \min(k^m_j, k^w_j)$.

3. There exists at least some $j$ such that $E(S_j) \leq \sum_{i=1}^{l} p_i E(S_i)$ and $k^w_j \leq k^*_j \leq k^m_j$. Similarly, there exists at least some $j'$ such that $E(S_{j'}) \geq \sum_{i=1}^{l} p_i E(S_i)$ and $k^w_{j'} \geq k^*_j \geq k^m_{j'}$.

Proof:

1. We need to show that

$$E(W^p(k)) = \frac{1}{2} \sum_{i=1}^{l} p_i E(S_i) + \frac{m}{2n} \sum_{i=1}^{l} p_i^2 E(S_i) n \geq \frac{1}{2} \sum_{i=1}^{l} p_i E(S_i) + \frac{m}{2n} \left( \sum_{i=1}^{l} p_i \sqrt{E(S_i)} \right)^2 .$$

Using the Cauchy-Schwarz inequality, we have:

$$\left( \sum_{i=1}^{l} p_i \sqrt{E(S_i)} \right)^2 = \left[ \sum_{i=1}^{l} \left( p_i \sqrt{E(S_i)} \right) \sqrt{\frac{n}{k_i}} \right]^2 \leq \left( \sum_{i=1}^{l} \frac{p_i^2 E(S_i) n}{k_i} \right) \left( \sum_{i=1}^{l} \frac{k_i}{n} \right) .$$

The equality holds if and only if $p_i \sqrt{E(S_i)} \sqrt{\frac{n}{k_i}} = c \sqrt{\frac{n}{k_i}}$ for all $1 \leq i \leq l$ with a common $c \neq 0$, which together with $\sum_{i=1}^{l} k_i = n$ leads to $c = \sum_{i=1}^{l} p_i \sqrt{E(S_i)}$ and $k = k^*$. 

2. By virtue of the Cauchy-Schwarz inequality,

$$\sum_{i=1}^{l} p_i \sqrt{E(S_i)} = \sum_{i=1}^{l} \sqrt{p_i \sqrt{E(S_i)}} \leq \sqrt{\sum_{i=1}^{l} p_i E(S_i)} .$$

If

$$E(S_j) \leq \sum_{i=1}^{l} p_i E(S_i) ,$$

we have

$$k^*_j = \frac{p_j \sqrt{E(S_j)}}{\sum_{i=1}^{l} p_i \sqrt{E(S_i)}} n \geq \frac{p_j \sqrt{E(S_j)}}{\sqrt{\sum_{i=1}^{l} p_i E(S_i)}} n = \frac{p_j E(S_j)}{\sum_{i=1}^{l} p_i E(S_i)} n = k^w_j ,$$

and by Theorem 2

$$k^m_j \geq k^w_j .$$

Similarly, if

$$E(S_j) \geq \sum_{i=1}^{l} p_i E(S_i) ,$$

7
we have

\[ k_j^* = \frac{p_j \sqrt{E(S_j)}}{\sum_{i=1}^{l} p_i \sqrt{E(S_i)}} n \geq \frac{p_j \sqrt{\sum_{i=1}^{l} p_i E(S_i)}}{\sqrt{\sum_{i=1}^{l} p_i E(S_i)}} n = k_j^m \]

and

\[ k_j^m \leq k_j^w. \]

In general, we have \( k_j^* \geq \min(k_j^m, k_j^w) \).

3. One side of the inequalities were shown in 2). For the remaining inequalities, consider the fact

\[ \sum_{j=1}^{l} k_j^* = \sum_{j=1}^{l} k_j^w = \sum_{j=1}^{l} k_j^m = n, \]

which would lead to a contradiction if the inequalities did not hold.

We should note that the optimal allocation \( k^* \) is only feasible if it leads to integer-valued \( k_j^* \)'s. If not, the optimal allocation would have to be obtained via a search over the discrete space of feasible values for the variables \( k_i \). Note that the function \( E(W^p(k)) \) is jointly convex in the variables \( k_i \), which would facilitate such a search.

**Corollary 4** The allocation defined by \( k^* \), if feasible, provides an overall expected time in system that is at least as small as the one provided by the unpartitioned system.

**Proof:** The result follows from the fact that

\[ E(W^p(k^*)) = \min_{\sum_{i=1}^{l} k_i = n} E(W^p(k)) \leq E(W^p(k^w)) = E(W^u), \]

where the last equality is due to Theorem 1.

The expression of the optimal allocation \( k^* \) is reminiscent of the workload-proportional allocation, except that the effect of service times is attenuated by the square root factor. This of course favors customer classes with relatively short service times and, as we can see, they receive a higher allocation of servers than they would under the workload-proportional allocation. This comes at the expense of customer with relatively long service times who receive a smaller allocation of servers; nevertheless, these customers are guaranteed to receive an allocation that is at least greater than or equal to the one they would receive under a mix-proportional allocation. In general, regardless of customer class, \( k_j^* \geq \min(k_j^m, k_j^w) \) is always satisfied.
The workload- and mix-proportional allocations can be viewed as two extreme forms of allocation, one accounting for service times, the other ignoring service time differences and accounting only for the relative population of each class. The optimal allocation accounts for service times but not to the extent that the workload-proportional allocation does. Because of this, one might assume that an ordering among the three allocations always holds. However, this is not true in general, although it is so for the important case of a system with two classes (this case is important because in practice customers are often partitioned into two classes). For this case, as a consequence of result 3 in Theorem 3, we have $k_1^w \leq k_1^* \leq k_1^m$ and $k_2^w \geq k_2^* \geq k_2^m$.

5. On the Benefit of Partitioning

In this section, we illustrate using a simple example the relative benefit that could be realized from partitioning. We consider a system with $l$ classes with equal populations, so that $p_i = 1/l$ for $i = 1, ..., l$. The means of the service times of different classes are uniformly distributed over the range $[M - x, M + x]$, where $x < M$ and $M > 0$. This implies that $E(S) = \sum_{i=1}^{l} p_i E(S_i) = M$ and $E(S_i) = M - x + 2x(i - 1)/(l - 1)$. For example if $M = 10$, $x = 3$ and $l = 4$, then $E(S_1) = 7$, $E(S_2) = 9$, $E(S_3) = 11$ and $E(S_4) = 13$. Using this construction, we can compare systems with the same overall mean $M$ but different variability in mean service times by varying $x$ and $l$ as shown in Figure 1.

Figure 1 shows the percentage decrease in expected time in system that results from optimal partitioning relative to no partitioning, as a function of standard deviation of service time means. We do so for different values of $l$ and for different values of standard deviation in the service time means (obtained by varying the parameter $x$). The curves from longest to shortest are for $l = 2, 3$ and 4 respectively. Other parameters are $M = 10$, $n = 10$ and $m = 500$. The following observations can be made.

1. The percentage decrease in expected time in system due to partitioning can be significant, up to 50 percent in the cases shown.

2. The improvement is increasing in the variability among the means of service times of the different demand classes.

3. For a given level of variability, the improvement is increasing in the number of demand classes.
Figure 1: Percentage Decrease in Expected Time in System due to Partitioning

The first observation is not surprising since the improvement that could be achieved with partitioning is not theoretically bounded but rather depends on the differences among service time means of the different demand classes. The fact that the improvement is not bounded can be seen from the limit

$$\delta = \lim_{m \to \infty} \frac{E(W^p(k^*))}{E(W^u)} = \frac{\left(\sum_{i=1}^{l} p_i \sqrt{E(S_i)}\right)^2}{E(S)}.$$  

It is easy to construct examples where $\delta$ is arbitrarily small and even approaches zero. The second observation is also consistent with intuition. The more variability in mean service times there is among different classes, the more valuable it is to separate these classes (by protecting those with short service times from those with long service times). The third observation is due to the fact that, everything else being equal, dividing customers in more classes is always desirable since we can always choose to treat two or more demand classes in the same way in terms of server assignments (which would be equivalent to merging them in a single class).

6. Concluding Comments

In this paper, we presented a model for studying the partitioning of servers during a rush hour demand regime. We set out to answer three basic questions: (1) is partitioning beneficial to
the system? (2) is it equally beneficial to all customer classes? and (3) if it is implemented, what is an optimal partition? We found that partitioning can be indeed beneficial to the system and this benefit can be significant. However, we also found that this benefit is realized only at the expense of one or more customer classes. In fact, we showed that it is impossible for all customer classes to benefit from partitioning. We showed that there is an optimal way to partition servers and provided via simple closed form expressions a characterization of the optimal partitioning. Because our main results are independent of assumptions about service time distributions and number of customers that arrive during rush hour, they are potentially useful for a wide range of applications. We expect our results to be particularly useful in settings where the rush effects are pronounced and where steady state analysis using alternative models, such as queueing models, is not applicable.

In this paper, we have assumed that customer classes are exogenously determined. A potential future extension of our model is to consider jointly the partitioning of servers among customer classes as well as the classification of customers into classes. This would entail jointly determining the number of customer classes, the range of service times associated with each customer class, and then the allocation of servers to classes. For example, in a supermarket environment, this would mean determining whether or not to have express lanes (i.e., dedicated lanes based on the number of items purchased by customers), the range of number of items to allow in each type of lane, and the number of lanes to have of each type.

Acknowledgments

The research of the first author was carried out while he was at the University of Minnesota at Minneapolis.

References


